

REMARKS

Claims 1, 14, 27 and 41 have been amended. Claims 1-14 and 16-53 remain pending in the application. Reconsideration is respectfully requested in light of the following remarks.

Claims Objected To But Otherwise Allowable:

Claims 13, 26, 40 and 53 were objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form. Applicants respectfully thank the Examiner for consideration of these claims. However, as discussed below, Applicants believe that the independent claims are allowable as currently written.

Section 103(a) Rejection:

The Examiner rejected claims 1-12, 14, 16-25, 27-39 and 41-52 under 35 U.S.C. § 103(a) as being unpatentable over Lucassen et al. (U.S. Publication 2003/0023953) (hereinafter “Lucassen”) in view of Green et al. (U.S. Publication 2002/0104067) (hereinafter “Green”). Applicants respectfully traverse this rejection for at least the following reasons.

Regarding claim 1, contrary to the Examiner’s assertion, Lucassen in view of Green does not teach or suggest a dynamic component generator configured to *receive a new set of requirements for the application; determine whether the new set of requirements includes changes from the initial set of requirements for the same application; and if the new set of requirements includes changes from the initial set of requirements, generate a second dynamic component to replace the first dynamic component.* The Examiner cites Lucassen, paragraphs 108, 109 and 123, where Lucassen describes dynamically generating interaction logic and presentation layers and customization at runtime. However, nowhere does Lucassen describe a dynamic

component generator *determining whether a new set of requirements for the application includes changes from an initial set of requirements for the same application.*

Lucassen teaches an application development system for multi-channel applications. Lucassen describes applications that allow a user to interface in parallel with same information via multiple channels and user interfaces, such as voice and graphics. Lucassen describes an interaction-based application framework utilizing different programming layers, such as a business logic layer, interaction logic layer, and customization layer, to specify the multi-channel applications. Lucassen's system includes an interaction manager for generating a presentation layer (Lucassen, Abstract, paragraphs 41, 59, 67, and 105 – 109). Specifically, Lucassen states, "the interaction manager 57 receives the interaction logic layer 53 and the customization meta-data 54 and generates functional or customized presentations for a particular delivery context." Lucassen further teaches that the interaction logic layer is "an abstract description of an application that describes how a user can interact with the application" (paragraph 107) and that the customization layer includes metadata associated with the interaction logic layer to optimize the presentation that will be generated by an adaptation process for a particular delivery context (e.g. voice or graphic). Lucassen further teaches that developers use a MVC-based editor/IDE development tool including a model editor for programming the interaction logic and customization layers (paragraphs 131,134 and 137). Thus, Lucassen's application generates a presentation from a developer-generated interaction logic layer and (developer-generated) customization meta-data.

Lucassen teaches that to change the presentation views, a developer would use the development tool to "access, edit and visualize the interaction logic and customization meta-data representation" (paragraph 137). After the developer modifies the underlying interaction logic and customization meta-data, Lucassen's application would generate new, different presentations.

Lucassen's system does not include a dynamic component generator configured to determine whether a new set of requirements for the application includes changes from

an initial set of requirements for the application. Instead, Lucassen teaches that new developer-generated interaction logic layers are used to generate new presentation layers. Thus, when a developer creates a new interaction logic layer, Lucassen's system will generate a new presentation layer accordingly. However, Lucassen's system does not include any dynamic component generator determining whether a new set of requirements for the application includes changes from an initial set of requirements for the application. Generating a new presentation layer based on a new interaction logic layer does not disclose or anticipate a dynamic component generator configured to determine whether a new set of requirements includes changes from an initial set of requirements, as recited in Applicants' claim.

Nor does Green, whether considered alone or in combination with Lucassen, teach or suggest a dynamic component generator determining whether a new set of requirements for the application includes changes from an initial set of requirements for the application. As explained in further detail below, the requirements described in Green are in regard to creating a new application. Green does not teach a dynamic component generator determining whether a new set of requirements for an existing application includes changes from an initial set of requirements for the application.

Further regarding claim 1, contrary to the Examiner's assertion, Lucassen in view of Green fails to teach or suggest *receive a new set of requirements for the application; determine whether the new set of requirements includes changes from the initial set of requirements; and if the new set of requirements includes changes from the initial set of requirements, generate a second dynamic component to replace the first dynamic component in the application, wherein the second dynamic component is configured to function according to the new set of requirements; wherein the dynamic component generator is configured to generate the second dynamic component to replace the first dynamic component by modifying or overwriting the first dynamic component.* The Examiner admits that Lucassen does not specifically disclose determining whether the new set of requirements includes changes from the initial set of requirements and generating a second dynamic component to replace the first dynamic component if the

new set of requirements includes changes from the initial set of requirements, wherein the dynamic component generator is configured to generate the second dynamic component to replace the first dynamic component by modifying or overwriting the first dynamic component, and relies on Green to teach these limitations. The Examiner submits that Green teaches these limitations in FIG. 5, paragraph [0060], paragraph [0064] and paragraph [0150]. However, Green is directed to an MVC framework that may be used to generate new applications by assembling them from components and tiers included in the framework.

Contrary to the Examiner's suggestion, the cited passages of Green do not describe receiving new requirements for an application that already exists, as required by Applicants' claims, but receiving requirements for a new application. For example, paragraph [0150] states, in its entirety:

In the operation of the preferred embodiment, referring generally to FIG. 6, to create a software application a set of application requirements is determined 70, either manually, heuristically, automatically, or by any combination thereof. Using predetermined N-tier architecture rules and optional wizards, a system designer determines 72, 74 a list of required models 31 and software components 20 to satisfy the application requirements. The list of required models 31 and software components 20 are logically grouped 72 into one or more packages 42 and the packages 42 associated with tiers 30. (emphasis added.)

In response to receiving requirements for a new application, the system of Green determines if the requirements can be met using the existing components 20 and tiers 30 of the development framework, or if the components and/or tiers must be modified and/or if new ones must be created in order to meet the requirements of the new application. If the requirements for the new application cannot be met using the existing components and/or tiers, changes will be made, as illustrated in FIG. 5 and described in paragraph [0060]:

The system implemented is put into production 52 and periodically reviewed for adjustments that may be necessary 54. If any tier 30 is determined to be in need of adjustment 56, it can be removed or otherwise modified 58. As additional requirements arise 60, new software components 20 are created, existing software components 20 modified 62,

64, or a combination thereof. Tiers 30 may be added, modified, or deleted 66 as application requirements dictate. (emphasis added.)

As described in paragraph [0060], it is the application development system, not an existing application, which is updated in response to receiving requirements for a new application. **When Green discusses modifying existing software components at paragraph [0060], Green is referring to components in the application development system, not components of an existing application.** The new application may then be created by assembling it from any combination of old, new, or modified components 20 and/or tiers 30. There is nothing in any of the Examiner's citations or elsewhere in Green that teaches or suggests *the dynamic component generator is configured to generate the second dynamic component to replace the first dynamic component in the application... by modifying or overwriting the first dynamic component* (i.e., in an application that was created according to a initial set of requirements for the application), in response to receiving a new set of requirements for that same application. Therefore, Lucassen in view of Green does not teach or suggest the above-referenced limitations of claim 1.

Applicants remind the Examiner that to establish a *prima facie* obviousness of a claimed invention, all claim limitations must be taught or suggested by the prior art. *In re Royka*, 490 F.2d 981, 180 U.S.P.Q. 580 (C.C.P.A. 1974), MPEP 2143.03. As discussed above, the cited references, taken separately or in combination, do not teach or suggest all the limitations of claim 1.

For at least the reasons above, the rejection of claim 1 is unsupported by the cited art and removal thereof is respectfully requested.

Claims 14, 27, and 41 include limitations similar to those discussed above regarding claim 1 and so the arguments presented above apply with equal force to these claims as well.

Applicants assert that numerous ones of the dependent claims recite further distinctions over the cited art. Applicants traverse the rejection of these claims for at

least the reasons given above in regard to the claims from which they depend. However, since the rejections have been shown to be unsupported for the independent claims, a further discussion of the dependent claims is not necessary at this time. Applicants reserve the right to present additional arguments.

CONCLUSION

Applicants submit the application is in condition for allowance, and prompt notice to that effect is respectfully requested.

If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5681-08800/RCK.

Respectfully submitted,

/Robert C. Kowert/

Robert C. Kowert, Reg. #39,255
Attorney for Applicants

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: May 29, 2007